Express Mail No.: EL822886561US

PATENT APPLICATION
ATTORNEY DOCKET NO. AHC-001
(5537/3)

PRACTICE MANAGEMENT AND BILLING AUTOMATION SYSTEM

Related Applications

[0001] This application claims priority to U.S. provisional patent application, serial number 60/223,235, filed on August 4, 2000, and entitled "Practice Management and Billing Automation System," the entire disclosure of which is hereby incorporated by reference.

Field of the Invention

[0002] The invention relates generally to the management of medical practices and more particularly to a medical practice management and billing automation system.

Background of the Invention

[0003] High administrative costs for filing and processing health insurance claims have typically been the bane of the health insurance industry for decades. Insurance claims include various information associated with a patient, such as the patient's insurance eligibility for a particular medical procedure. An unfortunate reality of today's healthcare industry typically is that billions of dollars are wasted in the processing of flawed insurance claims. Additionally, the processing of these insurance claims often wastes valuable time, typically resulting in an average revenue recognition cycle of over seventy days.

[0004] As a specific example, a medical professional (e.g., a receptionist) can verify the insurance eligibility of a patient by calling the particular insurance company. Alternatively, the receptionist can request eligibility information for the patient using a web portal electronically connected to the insurance company's web page. The receptionist types information about the patient into the web portal, transmits the information to the insurance company's web server, retrieves a response, and manually enters this information into a computer of the medical practice. Due to the large number of steps involved for this task and also due to the heavy workload frequently placed on the professionals performing these tasks, data entry errors often occur. These errors typically slow the process of successfully submitting an acceptable claim to the insurance company. For instance, the medical professional (e.g., receptionist) can forget to initiate an eligibility check, which can lead to significant billing and claim processing problems.

[0005] Other areas associated with the management of a medical practice, such as the claim acknowledgement area, often experience the same or similar problems. Particularly, once a medical practice submits an insurance claim to the insurance company, a medical professional typically has to sift through numerous claim acknowledgement reports to determine if the claim has reached its correct destination. This is a time-intensive, manual process that can be inundated with human error. These errors include, for example, errors in the comparison of the content of a report to the claim submission records stored in a medical practice management system. Another aspect of the process that often has similar problems includes the process of checking on the status of the claim as the claim traverses the insurance company system.

[0006] Given the extent of wasted time and money associated with the process, there exists a need to manage a medical practice in a more efficient manner and provide insurance claims with fewer or no errors.

Summary of the Invention

[0007] The present invention automatically and repeatedly interacts with an insurance company system and/or applies rules to efficiently manage a medical practice and provide insurance claims with a reduced number of errors. In one aspect, the invention includes a method for managing a medical practice. The method includes communicating with a medical practice client user interface over a first communications network and a payor server over a second communications network. The method also includes receiving information associated with an event related to a patient from the medical practice client user interface and/or the payor server and performing one or more tasks associated with the event. Additionally, the information associated with the event is used to create an insurance claim following completion of the task(s). The invention also automatically and repeatedly interacts with the information associated with the event in connection with the performed tasks by applying one or more rules in a set of rules and/or performing transactions with the payor server.

[0008] In one embodiment, the method includes the step of verifying the information before, during, and/or following performing the task(s) associated with the event. The method can also include receiving an error notification and performing a correcting action, which can include transmitting an error message denoting an error to the medical practice.

[0009] In another aspect, the invention includes a medical practice management system. The medical practice management system includes a medical practice client user interface, a payor server, and a medical practice management server computer. The medical practice client user interface communicates with a patient and the payor server communicates with a payor organization. The medical practice management server computer receives information associated with an event related to a patient from the medical practice client user interface and/or the payor server. The medical practice management server additionally includes a workflow processing engine, a rules engine, and an intelligent transactions relationship module.

[0010] The workflow processing engine performs one or more tasks associated with the event and the rules engine repeatedly and automatically interacts with the information associated with the event by applying one or more rules in a set of rules to the information in connection with one or more of the tasks. The intelligent transactions relationship module repeatedly and automatically interacts with the information associated with the event by performing transactions with the payor server in connection with the performance of one or more tasks.

[0011] In one embodiment, the workflow processing engine further comprises a verifier to verify the information before, during, and/or after performing the tasks associated with the event.

Brief Description of the Drawings

[0012] The above and further advantages of this invention may be better understood by referring to the following description in conjunction with the accompanying drawings. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the invention.

[0013] Fig. 1 illustrates a block diagram of an embodiment of a medical practice management system according to the present invention.

[0014] Fig. 2A illustrates a more detailed block diagram of an embodiment of a medical practice management server of the medical practice management system according to the present invention.

[0015] Fig. 2B illustrates a block diagram of an embodiment of the functions performed by a rules engine according to the present invention.

[0016] Fig. 3A illustrates a flow diagram of an embodiment of the steps performed by the medical practice management system according to the present invention.

[0017] Fig. 3B illustrates an embodiment of a patient workflow diagram performed by the medical practice management system before a patient visits a medical practice according to the present invention.

[0018] Fig. 3C illustrates an embodiment of the steps performed by the medical practice management system to determine patient eligibility.

[0019] Fig. 3D illustrates an embodiment of the steps performed by the medical practice management system to determine patient referral / prior authorization information.

[0020] Fig. 3E illustrates an embodiment of a patient workflow diagram performed by the medical practice management system while a patient visits a medical practice according to the present invention.

[0021] Fig. 3F illustrates an embodiment of a billing workflow diagram performed by the medical practice management system according to the present invention.

[0022] Fig. 4 illustrates an exemplary embodiment of a patient registration screen according to the present invention.

[0023] Fig. 5 illustrates an exemplary embodiment of a patient check-in screen according to the present invention.

[0024] Fig. 6 illustrates an exemplary embodiment of a patient check-out screen according to the present invention.

[0025] Fig. 7A illustrates an exemplary embodiment of a claim entry screen according to the present invention.

[0026] Fig. 7B illustrates an exemplary embodiment of an advanced claim entry screen according to the present invention.

[0027] Fig. 7C illustrates an exemplary embodiment of a claim review screen according to the present invention.

[0028] Fig. 7D illustrates an exemplary embodiment of a claim edit screen according to the present invention.

[0029] Fig. 7E illustrates an exemplary embodiment of an explanation portion of the claim edit screen shown in Fig. 7D according to the present invention.

[0030] Fig. 8A illustrates an exemplary embodiment of the communications performed by several of the components of the medical practice management system according to the present invention.

[0031] Fig. 8B illustrates an exemplary embodiment of the tasks performed by the workflow processing engine and a referral rules engine according to the present invention.

[0032] Fig. 8C illustrates an exemplary embodiment of the tasks performed by a claim rules engine 60 and the workflow processing engine according to the present invention.

[0033] Fig. 8D illustrates an exemplary embodiment of the tasks performed by an insurance package selection rules engine and the workflow processing engine according to the present invention.

[0034] Fig. 8E illustrates an exemplary embodiment of the flow of information into and from the insurance information database according to the present invention.

Detailed Description

[0035] Fig. 1 illustrates a block diagram of an embodiment of a medical practice management system 5 that includes a medical practice client computer (or medical practice client) 10, a medical practice management server (or server) 14, and a payor server computer (or payor server) 18. The medical practice client 10 is in communication with the medical practice management server 14 over a medical practice client-server communication path 22 and passes through a first communications network (or medical practice client-server network) 26. The medical practice management server 14 is also in communication with the payor server 18 over a payor server communication path 30 and passes through a second communications network (or payor server network) 34. It should be noted that Fig. 1 is an exemplary embodiment intended only to illustrate, and not limit, the invention.

[0036] The medical practice client-server network 26 and the payor server network 34 can be a local-area network (LAN), a medium-area network (MAN), or a wide area network (WAN) such as the Internet or the World Wide Web (i.e., web). In one embodiment, the medical practice client-server network 26 (e.g., the medical practice client-server communication path 22) supports secure communications. In a further embodiment, communications occur after a medical care provider's, or user's, password is verified by the medical practice management server 14. Exemplary embodiments of the communication paths 22, 30 include standard telephone lines, LAN or WAN links (e.g., T1, T3, 56kb, X.25), broadband connections (ISDN, Frame Relay, ATM), and wireless connections. The connections over the communication paths 22, 30 can be established using a variety of communication protocols (e.g., TCP/IP, IPX, SPX, NetBIOS, Ethernet, RS232, and direct asynchronous connections).

[0037] The medical practice client 10 can be any personal computer (e.g., 286, 386, 486, Pentium, Pentium II, Macintosh computer), Windows-based terminal, network computer, wireless device, information appliance, RISC Power PC, X-device, workstation, mini computer, main frame computer, personal digital assistant, or other computing device that has a windows-based desktop, can connect to a network and has sufficient persistent storage for executing a small, display presentation program. Windows-oriented platforms supported by the medical practice client 10 can include, without limitation, WINDOWS 3.x, WINDOWS 95, WINDOWS 98, WINDOWS NT 3.51, WINDOWS NT 4.0, WINDOWS 2000, WINDOWS CE, MAC/OS, Java, and UNIX. The medical practice client 10 can include a visual display device (e.g., a computer monitor), a data entry device (e.g., a keyboard), persistent or volatile storage (e.g., computer memory) for storing downloaded application programs, a processor, and a mouse.

[0038] The medical practice client 10 includes a medical practice client user interface 36. The interfaces 36, 40 can be text driven (e.g., DOS) or graphically driven (e.g., Windows). In one embodiment, the medical practice client user interface 36 is a web browser, such as Internet Explorer™ developed by Microsoft Corporation (Redmond, WA), to connect to the medical practice client-server network 26. In a further embodiment, the web browser uses the existing Secure Socket Layer (SSL) support, developed by Netscape Corporation, (Mountain View, CA) to establish the medical practice client-server network 26 as a secure network.

[0039] The medical practice management server 14 and the payor server 18 can be any personal computer described above. In one embodiment, the medical practice management server 14 hosts one or more applications 44 that the medical practice client 10 can access. Moreover, the payor server 14 can host one or more applications 45 that the medical practice management server 14 can access. In another embodiment, the medical practice management server 14 (and/or the payor server 18) is a member of a server farm, which is a logical group of one or more servers that are administered as a single entity. In the embodiment shown, the server farm includes the server 14, a second server 48, and a third server 52.

[0040] In a further embodiment, a second medical payor server computer (not shown) communicates with the server 14 through the payor server network 34.

[0041] In one embodiment, a medical care provider uses the medical practice client 10. Examples of the medical care provider include, but are not limited to, medical physicians, medically trained individuals, medical specialists, medical experts, receptionists, and the like. The medical practice client 10 is typically located in a medical practice. In one embodiment, the medical practice is the office of the medical care provider (e.g., a doctor's office), a hospital, other facilities providing medical treatment, and the like. Further, in one embodiment, a payor organization, or payor, uses the payor server 18. Although also referred to below as an insurance company, example embodiments of a payor organization also include, but are not limited to, health maintenance organizations (HMOs). More specifically, examples of payor organizations include, without limitation, Century Health and Benefits, HMO Blue, Harvard Pilgrim Health Care, MassHealth, Medicare, Neighborhood Health Plan, Tufts Associated Health Plan, United Healthcare, and the like.

[0042] Referring to Fig. 2A, the medical practice management server 14 includes a workflow processing engine 56, a rules engine 60, and an intelligent transactions relationship (ITR) module 68. In one embodiment, the rules engine 60 includes a rules database interface 64 and a rules database 66. In one embodiment, the engines 56, 60 and/or the ITR module 68 are software modules located within the medical practice management server 14. In another embodiment, one or more of the engines 56, 60 and/or the ITR module 68 are externally located from the server 14 and communicate with the server 14.

[0043] In one embodiment, the workflow processing engine 56 is a software application that controls and manages the features and functions of the medical practice management system 5. The workflow processing engine 56 and the medical practice client 10 communicate over the medical practice client-server network 26. In operation, the medical practice client 10 transmits a medical care provider request containing information to the medical practice management server 14 using, for example, a common gateway interface (CGI) request. For example, when registering a new patient, a medical care provider operating the medical practice client 10 enters the relevant patient information on a patient registration template that the workflow processing engine 56 delivered to the medical practice client user interface 36.

[0044] The workflow processing engine 56 also checks the structure and composition of information entered by a medical care provider at the medical practice client 10 to ensure that the information is correct (i.e., structure and/or composition). Examples of information entered by a medical care provider at the medical practice client 10 include the patient's address, phone number, medical history, insurance information, diagnosis and procedure codes, and the like.

[0045] The workflow processing engine 56 is additionally in communication with the rules engine 60. The rules engine 60 enables real-time application of "rules" stored in the rules database 66. Described in more detail below with respect to Fig. 2B, a rule is coded logic that evaluates data and then performs an action.

[0046] The rules engine 60 can access and update information stored in the rules database 66 using the rules database interface 64. Although not shown in Fig. 2A, in another embodiment the rules database interface 64 is a software layer internal to the workflow processing engine 56. The rules database interface 64 can be, for example, an application program interface or a Component Object Model (COM) object, which was developed by Microsoft Corporation.

[0047] The rules database 66 and/or the rules database interface 64 may be written in a structured query language, such as SQL, developed by IBM Corporation (Armonk, NY). In one embodiment, the rules database interface 64 uses a Lightweight Directory Access Protocol (LDAP) to access information in the rules database 66. Additionally, the rules database 66 can be external to the server 14 or may be internally situated in the server 14.

[0048] The rules database 66 includes insurance company rules that define the appropriate format and content of clinical and claim information that the payor server 18 processes. In one embodiment, the rules are subdivided into various classes. For example, the rules are divided into rules that have universal applicability to all claims for a specified payor, rules that apply only to one or more specific insurance packages from among the variety of insurance packages that the payor offers to medical care providers, and rules that apply only to specific medical care providers who provide care under one or more specific insurance packages.

[0049] Typically, a trigger invokes the application of a particular rule. For example, the submission of an insurance claim for a first payor could invoke the rules engine 60 to apply particular formatting rules associated with the first payor to format the claim to the first payor's specification.

[0050] To ensure that the rules database 66 contains current rules, the rules database 66 is frequently updated. In one embodiment, individual payors transmit rule updates / creations to the medical practice management server 14 via their payor server 18. Rule specialists review the rules transmitted by the payor server 18 and subsequently update the rules database 66. In one embodiment, the rules specialist performs any and all updates to the rules database 66. Alternatively, the updating of the rules database 66 can be automated upon receipt of a rule transmission from the payor server 18 or the medical practice client 18.

[0051] Additionally, a medical care provider can submit information to the medical practice management server 14 for subsequent update of the rules database 66 based on the medical care provider's experience with one or more payors. In yet another embodiment, the rules database 66 is updated with the server's historical analysis of previously submitted claims, especially those that were denied, to identify the reasons for denial. The historical analysis of previously submitted claims can facilitate the development of new rules for the rules database 66.

[0052] Referring to Fig. 2B, the rules engine 60 may interact with several payors (and therefore several payor servers 18), such as a first payor 70, a second payor 74, and a third payor 78. The rules engine 60 receives information 82, such as an insurance claim, from the workflow processing engine 56. In one embodiment, the rules engine 60 determines the payor 70, 74, 78 that the information 82 will be submitted by, for instance, searching the information 82 for a payor field. Once the rules engine 60 determines the receiving payor 70, 74, 78, the rules engine 60 applies the appropriate rules that are stored in the rules database 66 for the particular payor 70, 74, 78 to the information 82.

[0053] For example, the rules engine 60 applies the rules to the information 82 for the first payor 70 and subsequently transforms the originally received information 82 into first information 82′ having a form acceptable to the first payor 70. Likewise, the rules engine 60 applies the rules to the information 82 for the second payor 74 and subsequently transforms the originally received information 82 into second information 82″ having a form acceptable to the second payor 74. The rules engine 60 performs the same process to the information 82 to format the information 82 into third information 82‴ acceptable to the third payor 78.

[0054] Referring again to Fig. 2A, in one embodiment the medical practice management server 14 also includes a patient information database 92 (shown in shadow) and an insurance information database 96 (shown in shadow). The workflow processing engine 56 stores all of the information associated with a registered patient in the patient information database 92. The patient information database 92 stores information associated with existing patients of the medical practice. This information can include the patient's address, phone number, zip code, height, weight, allergies, previous doctor(s), and the like. In one embodiment, the medical practice management server 14 indexes the patient information stored in the patient information database 92 by the patient name. In another embodiment, the server 14 indexes the patient information stored in the patient information database 92 with a patient identifier. The patient identifier can be a random number, a predetermined integer (such as a patient counter), the patient's zip code, the patient's phone number, and the like. The workflow processing engine 56 typically accesses the patient information database 92 using a patient information database interface 93.

[0055] Similarly, the workflow processing engine 56 can store all of the information associated with an insurance company in the insurance information database 96, such as the insurance company's address, the amount of insurance coverage for a particular patient, and the like. Moreover, the workflow processing engine 56 can access the insurance information database 96 using an insurance information database interface 97.

[0056] In operation, as the workflow processing engine 56 receives information from the medical practice client 10, the workflow processing engine 56 determines on a real time basis whether all of the required information has been provided and whether the information is in the correct format. In the event that there is a deficiency in the information, the workflow processing engine 56 alerts the medical care provider (e.g., receptionist), or user, for additional information. Alternatively, the workflow processing engine 56 corrects the defect.

[0057] For instance, if the rules engine 60 contains a rule about member identification formatting for a particular payor, the rules engine 60 determines the rule in the rules database 66 and communicates the information to the workflow processing engine 56. The workflow processing engine 56 communicates this information to the medical practice client 10 when a medical care provider (e.g., receptionist) is registering a patient. If the medical care provider (e.g., receptionist) errs, the medical practice management server 14 alerts the medical care provider (e.g., with a warning message) to correct the error. This enables medical care providers to generate claims with no errors (i.e., referred to below as clean claims) for the mutual benefit of the medical care provider and the payor. Additionally, the medical care providers can obtain the information associated with an alert while the patient is physically present.

[0058] The workflow processing engine 56 is also in communication with the ITR module 68. The ITR module 68 executes transactions sent to and received from the payor server 18. Thus, the majority of provider/payor transactions can be accomplished electronically, with little or no human intervention. Examples of these transactions include, without limitation, claim submittals, claim receipt acknowledgements, claim status checks, patient eligibility determinations, authorization and referral requests and grants, and remittance advice. For example, a predetermined number of days before a scheduled patient visit, the ITR module 68 automatically checks patient eligibility with the applicable payor identified during the patient registration process. After a patient visit and the completion of the claim template, the claim is submitted to the payor server 18 via the ITR module 68.

[0059] In one embodiment, upon receipt of an insurance claim, the payor client 18 transmits a confirmation back to the medical practice management server 14. Later, on a schedule determined by the medical care provider, the ITR module 68 checks the claim status and notifies the medical practice client 10 accordingly. After the ITR module 68 analyzes the claim and generates remittance advice, the ITR module 68 parses the electronic payment and allocates the payment among the individual charge line items for the services provided. Once the medical care provider approves the allocations, the payments are posted to the provider's accounts.

[0060] Although described above as individual components, the engines 56, 60 and the ITR module 68 can be combined into one component or any number of components. Similarly, the databases, 66, 92, 96 could also be combined into one database and can be external or internal to the server. In other embodiments, the patient information and/or the insurance information is stored on a disk, such as a compact disk or a ZipDrive, developed by Iomega Corporation (Roy, Utah).

[0061] The medical practice management system 5 performs operations in response to an event related to a patient. Although a patient visit is used hereinafter as the event, the event can also be an emergency phone call to the medical provider, an emergency visit to the medical provider, a "virtual" visit to the medical practice client 10 (i.e., on-line communications with the medical practice client 10, such as over the Internet), and the like.

[0062] The medical practice management server 14 receives information associated with the event related to a patient from the medical practice client 10 and/or from the payor server 18 over the respective network 26, 34. The medical practice management server 14 performs one or more tasks associated with the event and then uses the information associated with the event to create an insurance claim after the completion of the task(s). An example of the information associated with the event is the patient information. The medical practice management server 14 automatically and repeatedly interacts with the information associated with the event in connection with the performed tasks by applying one or more rules in a set of rules and/or by performing transactions with the payor server 18.

[0063] Referring to Fig. 3A, the workflow of the medical practice management system 5 can typically be broken down into two sub-categories: 1) the patient workflow and 2) the billing workflow. The patient workflow can be sub-divided into tasks performed by the medical practice management system 5 before the patient visit to the medical practice and tasks performed by the medical practice management system 5 during the patient visit to the medical practice.

[0064] The medical practice management server 14 performs particular tasks before the patient visits the medical practice (step 302). In one embodiment, at the end of each task within the group of tasks that the medical practice management server 14 performs before the patient visit, the medical practice management server 14 automatically verifies, or interacts with, all information related to the task that the medical practice management server 14 receives from the medical practice client 10 (step 304). In another embodiment, the medical practice management server 14 automatically verifies all information related to the task (step 304) in real-time (i.e., substantially simultaneous with the medical care provider entering the information into the medical practice client 10). The medical practice management server 14 automatically interacts with / verifies the information by applying a particular rule (i.e., stored in the rules database 66) and/or by communicating with the payor server 18 (via the ITR module 68). The type of interaction / verification that occurs can also depend on the type of information. Additionally, the interaction / verification with the information can include a separate workflow that involves multiple steps and processes. In particular, the interaction / verification may also include application of a set of rules, communication with a payor organization via the ITR module 68, checking the format of the information, and the like at any time (e.g., before, during, after) throughout one or more of the performed tasks.

[0065] For example, if the medical practice client 10 transmits a form that includes the patient's address and zip code, and the zip code entered into the form has six digits instead of five, the medical practice management server 14 (i.e., the workflow processing engine 56) determines that the zip code is incorrect (step 306) and subsequently performs a correcting action on the zip code (step 308). In one embodiment, the workflow processing engine 56 alerts the medical care provider that the zip code is incorrect. In a further embodiment, the workflow processing engine 56 alerts the medical care provider with a pop-up window on the user interface 36 of the medical practice client 10. In yet another embodiment, the workflow processing engine 56 alerts the medical care provider with a voice message stating that the zip code is incorrectly entered.

[0066] Although described above and below as alerting the medical provider via the medical practice client 10, the correcting action could be any step to help correct the problem. For example, the workflow processing engine 56 could highlight the error(s) on the screen in a predetermined color, such as yellow. In yet another embodiment, the workflow processing engine 56 automatically corrects the zip code. For instance, the workflow processing engine 56 uses the patient's name to search the patient information database 92 for the patient's zip code stored in the database 92. Additionally, the correcting action can be performed by the workflow processing engine 56, the rules engine 60, and/or the ITR module 68. Moreover, the correcting action can be performed with operator assistance, such as with the assistance of a rules specialist.

[0067] After completing the tasks associated with the patient before visiting the medical practice, the medical practice management server 14 begins performing tasks associated with the patient's visit to the medical practice at the start of the patient's visit (step 310). The workflow processing engine 56 automatically verifies and checks the entered information during / after the performance of each task (step 312). If the medical practice management server 14 determines an incorrect / inaccurate piece of data, the workflow processing engine 56 alerts the medical care provider of the inaccuracy (step 316). Following the patient's visit to the medical practice, the medical practice management server 14 performs the billing tasks associated with the services provided to the patient (step 318). As shown in steps 320-324, the workflow processing engine 56 automatically verifies the billing information and alerts the medical care provider if necessary. It should be noted that, at each stage of the process (e.g., tasks performed before the visit to the medical practice, tasks performed during the visit to the medical practice, tasks performed during the billing process), the medical practice management server 14 verifies and checks each piece of information entered without human intervention (e.g., the medical provider). The verification and checking can be through rule application or by communication with a payor server 10 (via the ITR module 68). This can be done prior to the submission of an insurance claim to enable later submissions of more accurate claims (e.g., flawless claims) to the payor (step 326). Alternatively, this can be done during or following claim submission to the payor server 18.

[0068] Further, although described as a linear flow of operations, the steps illustrated in the Figs. 3A – 3F can be performed at various times before, during, or after the performed tasks. The steps illustrated can also be performed simultaneously. Moreover, the information received in one particular step can be used in other steps in the same workflow or other steps in other workflows. For example, the medical practice management server 14 (i.e., one or more of its components 56, 60, 68) can retrieve information associated with a patient at any of the steps illustrated in any of the Figs. 3A – 3F and can repeatedly use the information again at any / all other steps throughout any / all of the workflows. Thus, the use of the information received at one particular step is dependent upon the location within a workflow as well as the information being accessed and/or entered into the medical practice client 10 and/or the medical practice management server 14 and/or the payor server 18. Consequently, Figs. 3A – 3F are intended only to illustrate, and not limit, the invention.

[0069] The workflow processing engine 56 receives a request for an appointment from the medical practice client 10 with respect to a particular patient (step 328). In one embodiment, the medical practice client 10 transmits the name of the individual to the medical practice management server 14 to determine whether the individual is an established patient or a new patient. The workflow processing engine 56 searches for the patient in the patient information database 92 (step 330) using the index of the patient information database 92 or a portion of the indexing field to determine if the patient is an established patient (step 332). For instance, the workflow processing engine 56 can search for the patient in the patient information database 92 using the patient's full name or a portion of the patient's name (e.g. last name).

[0070] If the workflow processing engine 56 does not find patient information associated with the patient in the patient information database 92, then the patient is a new patient. In one embodiment, the workflow processing engine 56 then transmits a message to the medical practice client 10 to request registration information from the medical practice client 10. In a further embodiment, the workflow processing engine 56 transmits a registration screen to the medical practice client 10 in which the medical care provider (e.g., receptionist) enters patient information associated with the patient into the medical practice client 10. An example of a registration screen 400 is illustrated in Fig. 4 and is described in more detail below.

[0071] The medical care provider (e.g., receptionist) enters the patient information associated with the patient into the medical practice client 10 and the medical practice client 10 transmits the patient information to the medical practice management server 14. The workflow processing engine 56 receives the patient information (step 334) and automatically verifies / checks the format and accuracy of the information entered at the medical practice client 10 (step 336). If the information has an incorrect format or is inaccurate (step 338), the workflow processing engine 56 can alert the medical care provider of the inaccuracy (step 340).

[0072] Before a patient's visit to the medical practice, the medical care provider also collects insurance information from the patient. The medical care provider client 10 transmits the information to the medical practice management server 14. The workflow processing engine 56 receives the insurance information associated with the patient (step 342) and automatically verifies the insurance eligibility of the patient (step 336). In one embodiment, the ITR module 68 verifies the insurance eligibility by communicating with the payor server 18 via the payor server network 34.

[0073] Also referring to Fig. 3C, when the ITR module 68 receives the eligibility information from the payor server 18, the workflow processing engine 56 parses the information (step 350) to determine whether or not the patient is eligible (step 352). If the patient is eligible, the workflow processing engine 56 marks the patient's eligibility, such as on the patient registration screen 400 (step 354). Further, if the patient is not eligible, the workflow processing engine 56 sets a particular software flag (e.g., not eligible flag) on the associated insurance policy (step 356). In a further embodiment, the workflow processing engine 56 searches the insurance plans to create a policy worklist for the medical provider (step 357). Moreover, the workflow processing engine 56 copies all of the claims associated with that patient to create a claim worklist (step 358). This worklist has to be cleared before the workflow processing engine 56 can use the ITR module 68 to transmit the claim to the payor server 18. Moreover, the workflow processing engine 56 may additionally parse additional information out of the eligibility information, such as the copayment amount (step 359). Although described above as the workflow processing engine 56 performing these tasks, any of the other components of the medical practice management server 14 (e.g., rules engine 60) can alternatively or collaboratively perform these tasks.

[0074] Referring again to Fig. 3B, in yet another embodiment, the workflow processing engine 56 stores insurance information for each established patient in the insurance information database 96 and updates the insurance information database 96 regularly at predetermined intervals. The workflow processing engine 56 can search the insurance information database 96 to verify the insurance eligibility of the patient (step 336). The workflow processing engine 56 can then alert the medical care provider about the patient's eligibility (step 340), such as with a message advising the medical care provider to call the insurance company directly to verify that the patient has valid insurance.

[0075] In one embodiment, the workflow processing engine 56 also receives referral information associated with the patient (step 344) and subsequently verifies the referral information to ensure that the patient is referred by the medical care provider quoted by the patient (step 336).

[0076] Also referring to Fig. 3D, to determine whether a patient's particular visit requires a referral, the rules engine 60 defines two constructs: a referral rule category and an appointment type class (step 360 and step 362). The rules engine 60 groups particular payor organizations into referral rule categories. The rules engine 60 also maps particular appointment types within medical practices onto an appointment type class. The workflow processing engine 56 then defines the intersection of a given referral rule category and a particular appointment type (step 364) to either require a referral / preauthorization, not require a referral / preauthorization, or maybe require a referral / preauthorization. The workflow processing engine 56 then classifies a patient into one of these intersection classifications (step 366) and transmits the referral classification (e.g., referral not required) to the medical practice client 10 (step 368). If a referral / preauthorization is required but does not exist, the workflow processing engine 56 does not submit the claim and waits for further updates / edits (i.e., sets the claim status to HOLD, as described further below). Thus, the result of the referral / preauthorization classification, like all other information / results, can be used various and numerous times throughout one or more workflows without limit to the invention.

[0077] In further embodiments, the ITR module 68 automatically initiates a referral inquiry with the payor server 18 if the workflow processing engine 56 determines that the patient requires a referral / preauthorization.

[0078] The medical care provider then schedules an appointment with the patient. As shown in Fig. 3B, the workflow processing engine 56 receives the proposed schedule appointment (step 346) and simultaneously verifies that the proposed schedule date is appropriate (step 336). For instance, if the medical care provider incorrectly enters that the scheduled date is in the year 2000 instead of the year 2001, the workflow processing engine 56 catches the typographical error and alerts the medical care provider about the error (step 340). In additional embodiments, the workflow processing engine 56 confirms the appointment (step 348) by displaying a confirmation screen on the medical practice client 10.

[0079] Subsequently, the patient visits the medical practice. Referring to Fig. 3E, the medical practice client 10 and the workflow processing engine 56 perform their respective tasks associated with patient check-in (step 376). In one embodiment, the medical care provider transmits a request for a check-in screen to the medical practice management server 14, such as by clicking on the name of the patient on the medical practice user interface 36. The workflow processing engine 56 transmits a check-in screen to the medical practice client 10 for input by the medical care provider. An exemplary check-in screen 500 is illustrated in Fig. 5 and is described in more detail below.

[0080] The medical care provider enters patient information into the check-in screen. The workflow processing engine 56 verifies all information entered into the check-in screen (step 370) without intervention from the medical provider. This verification can occur in real-time, i.e., while the medical care provider enters the information into the check-in screen, or after the medical care provider completes the patient check-in screen and submits it to the medical practice management server 14 (as described below). Further, in one embodiment, the server 14 (i.e., the workflow processing engine 56) does not accept a check-in screen until all of the required patient information for that patient is complete.

[0081] In one embodiment, the patient check-in screen enables the medical care provider to edit (e.g., update) the insurance information received during the tasks performed before the patient's visit. Upon an edit, the workflow processing engine 56 verifies the insurance information (step 378) to determine if any inaccuracies exist and can alert the medical care provider (step 374) upon a finding of an error.

[0082] The medical care provider then collects the copayment from the patient for the visit to the medical practice and enters the amount of the copayment into the check-in screen. In one embodiment, the medical care provider then submits the check-in screen to the medical practice management server 14. The workflow processing engine 56 receives the check-in screen (step 380) and, in one embodiment, verifies all of the information entered into the patient check-in screen (step 370).

[0083] The workflow processing engine 56 then transmits a check-out screen to the medical practice client 10 (step 382). An exemplary check-out screen 600 is illustrated in Fig. 6 and is described in more detail below. In one embodiment, the medical care provider then provides a billing slip to the patient for the services incurred.

[0084] Once the patient is prepared to leave the medical practice, the medical care provider and the workflow processing engine 56 perform check-out tasks (step 384). For instance, these check-out tasks include, without limitation, rescheduling an appointment, creating an appointment reminder, and creating a chart check. A chart check is a systematic way to keep track of patients needing follow up. At each of these tasks, the workflow processing engine 56 automatically verifies that the information is correctly entered (step 370) and alerts the medical practice client 10 if there is some sort of error (step 374). After the check-out task are complete, the medical care provider submits the check-out screen to the medical practice management server 14.

[0085] The workflow processing engine 56 then transmits a claim entry screen to the medical practice client 10 (step 386) after receiving the check-out screen. An exemplary claim entry screen 700, or claim entry form, is illustrated in Fig. 7A. Alternatively, the workflow processing engine 56 transmits an advanced claim entry form to provide additional features, such as the advanced claim entry form 732 shown in Fig. 7B. The medical care provider uses the claim entry screen to complete a patient charge entry for the particular patient who had just checked out of the medical practice.

[0086] The creation of a claim links the tasks performed in Fig. 3E by the medical practice management server 14 during the patient's visit with the tasks performed in Fig. 3F by the server

14 during the billing process. In one embodiment, the start of the creation of a claim occurs following check-out of the patient and continues into the billing workflow of the medical practice management system 5. Also referring to Fig. 3F, when the medical care provider completes the claim entry form, the medical care provider submits the claim entry form to the server 14.

[0087] The workflow processing engine 56 transmits the claim entry form (step 389), which ultimately becomes the claim, to the rules engine 60. The rules engine 60 "scrubs" the claim (step 390), or examines the claim for claim errors. Claim errors can include, without limitation, typographical errors, formatting errors (based on a format that each payor defines for their claims), incomplete information, and the like. As described above, the payor server 18 and/or the rule specialists can update the rules database 66 with new or updated rules. Thus, the rules engine 60 can apply different rules to a claim at different times, depending on if the rules database 66 is updated or changed during the life of the claim.

[0088] Upon creation of a claim, the workflow processing engine 56 assigns a claim status to the claim (step 392). In one embodiment, the claim status denotes the results of the scrubbing of the claim. The possible claim statuses upon creation of the claim are shown in the table below:

Claim Status	Description
DROP	Identifies claims that are ready to be billed
HOLD	Identifies claims that have not passed the scrubbing done by the rules engine due to simple errors
MGRHOLD	Identifies claims that have not to passed the scrubbing done by the rules engine due to more detailed claims errors, such as a missing provider number
CLOSED	Identifies claims that have a zero balance

[0089] The workflow processing engine 56 then transmits a claim review screen to the medical practice client 10 illustrating the errors in the claim. An exemplary claim review screen 756 is illustrated in Fig. 7C. In one embodiment, these errors must be resolved before the medical practice management server 14 can process the claim. Moreover, in one embodiment the workflow processing engine 56 assigns some sort of HOLD status (e.g., MGRHOLD) to the incorrect claim to denote that the claim cannot currently be transmitted to the payor server 18.

[0090] In another embodiment, the medical care provider enters charges for multiple patients that have visited the medical practice within a predetermined amount of time (e.g., on a particular day). This is also referred to as entering a batch submission of patient charges. In one embodiment, this option is available in the advanced claim entry form shown in Fig. 7B.

[0091] The workflow processing engine 56 then determines if the claim has its claim status set to DROP (step 394). If so, the workflow processing engine 56 submits the claim to the payor (i.e., payor server 18) using the ITR module 68 (step 396). In one embodiment, the workflow processing engine 56 also communicates with a central billing office (CBO) (not shown). The CBO generates and submits the claims to the payor. The CBO transmits the claim to the payor server 18 over the payor server network 34. In yet another embodiment, if the payor associated with the claim is not accessible via the payor server network 34, the CBO transmits a paper version of the claim to the payor. Once the claim is transmitted to the payor (via the payor server network 34 or on paper), the workflow processing engine 56 changes the status of the claim to BILLED. It should be noted that the workflow processing engine 56 could perform the tasks that the CBO performs.

[0092] In further embodiments, the workflow processing engine 56 places an alarm on the claim to enable a high level of control and management of the claims. Typically, a user of a billing system executes a report on the age of the claim and performs work on the claims that are older than a predetermined time. In one embodiment, the medical practice management server 14 includes this functionality. However, the medical practice management server 14 can alternatively use alarms with the claims to control and manage the claims.

[0093] The alarm placed on the claim could depend on the particular claim clearinghouse used and/or on the particular payor. Additionally, the workflow processing engine 56 determines in step 396 the time that the claim is submitted to the payor. If the ITR module 68 does not receive a response from the payor server 18 within a predetermined amount of time, the workflow processing engine 56 triggers the alarm. Upon the triggering of the alarm, the workflow processing engine 56 moves the claim into a claim inquiry grouping of claims. The claim inquiry grouping of claims are claims that must be followed up on by, for instance, the CBO or by the workflow processing engine 56. Additionally, the workflow processing engine 56 may also set the claim status to BILLED and may place another alarm on the claim. This alarm is a function of the insurance and a "kickreason". A "kickreason" is a code that the workflow processing engine 56 sets to describe what has to be done to the claim to resolve the claim. In further embodiments, this code is insurance-specific.

[0094] The workflow processing engine 56 and/or the ITR module 68 then performs claim follow-up tasks (step 398). In one embodiment, as part of the claims follow-up tasks, the workflow processing engine 56 transmits a claim edit screen to the medical practice client 10 to enable the medical care provider to correct the claims that have errors. An exemplary claim edit screen 768 is illustrated in Fig. 7D and is described in more detail below. In another embodiment, the claim edit screen includes an explanation portion to explain the claim errors to the medical care provider. An exemplary explanation portion 770 of the claim edit screen 768 is illustrated in Fig. 7E. The medical care provider updates the claim based on the errors denoted in the claim review screen 756 and/or the explanation portion 770 of the claim edit screen 768. In one embodiment, the rules engine 60 scrubs the claim again following the editing by the medical care provider and the workflow processing engine 56 assigns a DROP status to the claim if no errors are found.

[0095] Additionally, once the medical care provider starts to receive payments for the claims, the medical practice management server 14 applies these payments against the associated charges. More specifically, the workflow processing engine 56 "posts" the payment for the associated claim (step 399). Furthermore, the workflow processing engine 56 can then assign a claim status of CLOSED to the claim associated with the payment that has been posted.

[0096] Furthermore, in other embodiments, the medical practice management server 14 can generate, transmit, and display reports to the medical practice client 10 about the medical practice. For instance, the workflow processing engine 56 can provide a report illustrating the medical practice's accounts receivable by date of service. Alternatively, the workflow processing engine 56 can search the patient information database 92 to provide a report of the demographic makeup of the patient population of the medical practice.

[0097] Referring to Fig. 4, an exemplary registration screen 400 includes a patient registration information section 404 and a notes section 408. The patient registration information section

404 can include any information required by the medical practice to register the patient. Examples of patient registration information included in section 404 are, without limitation, the patient's name, address, social security number, phone number, employer, and the like. The notes section 408 includes a section for the medical provider (e.g., receptionist) to provide any sort of notes about the patient.

[0098] Referring to Fig. 5, a check-in screen 500 includes a view / cancel appointment section 504, an edit appointment information section 508, an insurance section 512, a verify and edit registration information section 516, and a collect patient payment section 520. These sections are for illustrative purposes only and can be altered and/or replaced to provide whatever information is required for checking a patient into the medical practice.

[0099] An exemplary check-out screen 600 is illustrated in Fig. 6. In one embodiment, the check-out screen 600 is divided into several sub-sections: the billing slip sub-section 602, the check-out actions 604, the receipt sub-section 608, and the collect patient payment subsection 612. If the medical care provider collects a billing slip from the patient, the medical care provider can select the receipt sub-section 608 and provide a receipt to the patient for the services incurred.

[0100] Fig. 7A illustrates an exemplary claim entry screen 700. The claim entry screen 700 includes a patient claim information section 704, a procedure section 708, a hint section 712, and a non-claim resulting button 716. In one embodiment, the patient claim information section 704 includes information such as the claim post date field 704a, a referring provider name field 704b, and a referral number field 704c. The procedure section 708 typically includes a location in which the medical provider can enter in a procedure number denoting the medical procedure performed on the patient during the patient visit. The medical provider can use the non-claim resulting button 716 to denote the appointment as not requiring the creation of a claim.

[0101] In further embodiments and as shown, the workflow processing engine 56 includes a create claim button 720 and an advanced claim button 724 in the claim entry screen 700. The medical provider hits the create claim button 720 after the claim entry form 700 is completely filled out. If a field of a section 704, 708, 712, 716 is empty and the medical provider hits the create claim button 720, the workflow processing engine 56 can denote that the claim is not complete to the medical provider. In another embodiment, the workflow processing engine 56 looks up patient information from the patient information database 92 and completes as many of the fields in the claim entry form 700 as possible.

[0102] In one embodiment, referring to Fig. 7B, if the medical provider hits the advanced claim button 724, the workflow processing engine 56 provides an advanced claim screen 732 to the medical practice client 10. The advanced claim screen 732 includes an advanced patient claim information section 736 and an advanced procedure section 740. The advanced patient claim information section 736 can include, for instance, a payor section 744 and an illness section 748. Moreover, the advanced procedure section 740 can include a range 750, 752 of dates in which a procedure was performed. It should be noted that any modifications to the advanced claim entry form 732 (e.g., additional information) can be included in the advanced claim entry screen 732. The advanced claim entry form 732 can also include a create claim button 754, as described above, and a simple claim button 755 to retrieve the claim entry screen 700 shown in Fig. 7A.

[0103] Fig. 7C illustrates a claim review screen 756 having a created claim summary section 758, a claim warnings section 760, a receipt section 762, and a payment section 764. The workflow processing engine 56 denotes warnings associated with the claim in the claim warnings section 760. Claim warnings could include messages such as "The supervising provider has no id number entered for the given insurance" or "The second procedure is a component of the first procedure. These codes may not be billed together." The receipt section 762 illustrates receipts that the workflow processing engine 56 can provide to the medical practice client 10. The payment section 764 includes information about payments received by the medical practice management server 14.

[0104] Referring to Fig. 7D and 7E, a claim edit screen 768 and a claim error explanation portion 770 of the claim edit screen 768 is shown. Alternatively, the claim edit screen 768 and the claim error explanation portion 770 are joined into one screen. The claim edit screen 768 and the claim error explanation portion 770 include a claim edit section 772, a charge history section 776, and a claim scrubbing error section 780 to enable efficient edits of the claims. A summary 784 of the charge history section 776 can also be included in the claim edit screen 768.

[0105] An exemplary embodiment of the communications performed by several of the components of the medical practice management system 5 is illustrated in Fig. 8A. A rules specialist, as described above, defines a claim rule category to classify particular rules into a particular group for application with particular claims (step 804). The rules specialist then consults with (e.g., converses with) the payors (e.g., the payor servers 18) to obtain information related to the payor organization (step 806). In one embodiment, the consultation is human communication. In another embodiment, the consultation is communication between the payor server 18 and the medical practice management server 14. Further, the rules specialist could also perform research on the payor organization to further fine-tune information associated with the payor organization, such as the claim rule category (step 806). This rules specialist then inputs this information into the rules engine 60.

[0106] In one embodiment, the rules engine 60 is further divided into categorized rules engines, such as a claim rules engine 60′ and a referral rules engine 60″. In one embodiment, the rule specialist inputs the claim rule information into the claim rules engine 60′ for future application with a claim (e.g., produced by the workflow processing engine 56).

[0107] Besides receiving information from the claim rules engine 60′ (e.g., upon application of claim rules), the workflow processing engine 56 also receives and transmits information from / to various other sources. For instance, the workflow processing engine 56 collects the patient information from the medical practice client 10 (step 808). Another step implemented by the workflow processing engine 56 is the collection of practice setup information (step 810). In one embodiment, the workflow processing engine 56 receives practice setup information from practice setup specialists. Examples of practice setup information include, without limitation, payor organization locations, departments located within payor organizations, payor identification information, and the like. In a further embodiment, the medical practice client 10 provides the practice setup information to the workflow processing engine 56. The workflow processing engine 56 also communicates with one or more payor servers 18, such as a first payor server 18′ and a second payor server 18″.

[0108] A rules specialist also defines a referral rule category and appointment type classes (steps 812 and 814). In one embodiment, the practice setup specialist (or medical practice client 10) retrieves the appointment type classes for use with the practice setup, as shown with arrow 816. The rules specialist can also consult with (e.g., converse with) and do research on the payor organizations to obtain information related to the payor organization (step 818). The rules specialist then transmits this information into a referral rules engine 60″, another categorized rules engine within the rules engine 60 that handles referral rules. When applicable, the referral rules engine 60″ applies these referral rules to information associated with the practice setup (step 810) and transmits the result(s) to the workflow processing engine 56.

[0109] The communications between the workflow processing engine 56 and the other components of the medical practice management system 5 (e.g., the claim rules engine 60′, the referral rules engine 60″, the payor servers 18′, 18″, and the medical practice client 10) can occur before, during, after, and between one or more tasks performed by the workflow processing engine 56. One or more of the communications can occur simultaneously with any of the other communications.

[0110] An embodiment of the tasks performed by the workflow processing engine 56 and the referral rules engine 60″ is illustrated in Fig. 8B. The rules specialist defines the referral rule category (step 824) and/or the appointment type classes (step 826) and stores the information in the rules database 66 (of the referral rules engine 60″). Additionally, the workflow processing engine 56 and/or the rules specialist can consult with the payor organization to obtain information related to the payor (step 828). Although shown as a linear workflow diagram, the above mentioned steps can be performed linearly or in parallel to one or more of the other steps.

[0111] The workflow processing engine 56 may perform one or more tasks in which the referral rules engine 60″ verifies the referral information (step 830). An embodiment of the tasks performed during a rules referral include the workflow processing engine 56 scheduling a patient (step 832), checking a patient into the medical practice 10 (step 834), checking a patient out of the medical practice client 10 (step 836), and entering (step 838) a claim for the patient.

[0112] During the verification of the referral information, the referral rules engine 60″ checks one or more of the appointment type class, referral rule category, and specialty intersection (step 840). Once the referral rules engine 60″ checks these items, the referral rules engine 60″ provides referral information to the workflow processing engine 56 (step 842).

[0113] An embodiment of the tasks performed by the claim rules engine 60′ and the workflow processing engine 56 is shown in Fig. 8C. As described above, a rules specialist defines the claim rule category (step 842), consults with and/or does research on the payor (step 844), and creates / modifies the claim rules (step 846). The specialist stores this information into the claim rules engine 60′.

[0114] The claims rules engine 60′ verifies the claim information and/or formats the claim (step 848) throughout numerous steps performed by the workflow processing engine 56. For instance, following the performance of the check-out tasks (step 850), the workflow processing engine 56 uses the information from the medical practice client 10 (and/or the payor information from the payor server 18) to produce the claim (step 852). The workflow processing engine 56 then determines if the claim is billable (step 854) based on the verification process performed by the claim rules engine 60′ in step 848. If not, the workflow processing engine 56 may edit the claim information based on information received from the claim rules engine 60′ (e.g., if the claim rules engine 60′ provides the workflow processing engine 56 with information relating to edits to be performed to make the claim billable) (step 856).

[0115] If the claim is billable, the workflow processing engine 56 transforms the claim into a format acceptable to the payor organization (i.e., the payor server 18) (step 858). As illustrated, the claim rules engine 60′ performs the verification and formatting task (step 848) throughout many of the tasks performed by the workflow processing engine 56. Moreover, information transmitted to the workflow processing engine 56 from the verification / formatting task (step 848) of the claim rules engine 60′ can be used during any one of the tasks performed by the workflow processing engine 56. The workflow processing engine 56 then submits the claim to the payor after formatting the claim to the payor's specifications (step 860).

[0116] During the verification and formatting process performed by the claim rules engine 60′, the claim rules engine 60′ checks the claim rule category and the service date of the claim (step 862) and uses this information to define claim information requirements and claim formatting rules (step 864). The claims rules engine 60′ verifies the claim information and formats the claim(s) (step 848) to facilitate acceptance by the provider. As shown by arrow 866, in one embodiment the claim rules engine 60′ checks the claim rule category and service date (step 862) in conjunction with (e.g., before, during, after) the verification process of the claim information and/or during the formatting of the claim (step 848). Additionally, the defining of the claim information requirements and claim formatting rules are used to verify and/or format the claim information, as illustrated with arrow 868.

[0117] In yet another embodiment and referring to Fig. 8D, the rules engine 60 further includes an insurance package selection rules engine 60‴. In one embodiment, an insurance specialist, similar to a rules specialist, enters the insurance information into the insurance package selection rules engine 60‴. In another embodiment, rather than having different specialists (e.g., rules specialist, insurance specialist) to enter the different types of information (e.g., claim rules, insurance rules) into the rules engine 60, one specialist could enter all of the information into the rules engine 60.

[0118] In one embodiment, the insurance specialist defines the insurance package selection category and may perform payor consultation and research to determine information about the payor (steps 870 and 872). The insurance specialist then creates / modifies the insurance package selection rules based on this definition and payor consultation (step 874) and transmits the information into the insurance package selection rules engine 60‴ (i.e., the rules database 66).

[0119] The workflow processing engine 56 receives the patient information from the medical practice client 10 (step 876) and determines which insurance package to use (step 878). This determination involves the insurance package selection rules engine 60‴ choosing an insurance package selection category (step 880). The choosing of the insurance package selection category further involves determining criteria for the insurance package selection (step 882), which can be based on the information stored in the rules database 66 (by the insurance specialist). Further, the insurance package selection rules engine 60‴ utilizes the specific rules and criteria to determine the insurance package (step 884) for the particular patient. The workflow processing engine 56 uses this information in step 878 and subsequently enters the policy information for the particular patient into a claim (step 886).

[0120] Fig. 8E illustrates the impact of the insurance information database 96 (as shown in Fig. 2A) on the rules engine 60 and the tasks performed by the workflow processing engine 56. A workflow 888 performed by the workflow processing engine 56 communicates with the rules engine 56 (e.g., the claim rules engine 60′, the referral rules engine 60″, the insurance package selection rules engine 60‴) at one or more times during one or more of the tasks in the workflow 888. The rules engine 60 applies one or more rules to the information used during the workflow 888. These rules originate from rule categories that, in one embodiment, are defined by specialists (step 890). In another embodiment, the rules engine 60 determines the rules categories. The rules categories are determined from information stored in the insurance information database 96.

[0121] In one embodiment, a specialist performs claims denial analysis (step 892) and/or payor setup (step 894) to produce the information for the insurance information database 96. Moreover, a specialist may also perform payor research to obtain more information on the payor organization to aid in attaining accurate rule categories (step 896).

[0122] In one embodiment, the insurance information database 96 stores all of the information associated with various payor organizations. Moreover, the information stored in the insurance information database 96 can be used for many functions of the medical practice management system 5. For example, in another embodiment the workflow processing engine 56 uses the information stored in the insurance information database 96 during claim submission.

[0123] Further, although some steps illustrated in Figs. 8A – 8E are described as a linear flow of operations, the steps can be performed at various times before, during, or after the performed tasks. The steps illustrated can also be performed simultaneously. Moreover, the information received in one particular step can be used in other steps in the same workflow or other steps in other workflows. Consequently, Figs. 8A – 8E are intended only to illustrate, and not limit, the invention.

[0124] The medical practice management system 5 thus increases the efficiency of a medical practice by automatically interacting with information associated with a patient using rules, databases, and/or communication links with payors during the workflow of the medical practice.

[0125] While the invention has been particularly shown and described with reference to specific preferred embodiments, it should be understood by those skilled in the art that various changes in form and detail can be made therein without departing from the spirit and scope of the invention as defined by the appended claims.